Rajarata University of Sri Lanka

Department of Computing

ICT 2206 – Internet Programming

Laboratory 01 – File Handling using C++
Time: 120 Minutes

Intended Learning Outcomes (ILOs):

By the end of this practical, students should be able to:

1. Create an application to create a file.
2. Write inputs from the keyboard to the file.
3. Explain the different file access methods and their uses.

Pre-Practical Preparation:

- **Reading/Resources**:
  - Students should refer to the lecture notes on computer programming and object-oriented programming.
  - Refer to https://www.w3schools.com/cpp/ for more information.

- **Preliminary Questions**:
  - What is a file?
  - What are the different "modes" that a file can be opened using an application?

Practical Objective:

To familiarise the students with the concept of files and with different techniques of file handling in C++

Practical Activity:

Step-by-Step Guide/Instructions:

Step 1: Reading a text string from the command prompt

- **Objective**: Understand different techniques for reading a string.

Instructions:

1. Open any text editor.

2. Type the following C++ code:

```
#include <iostream>
#include <string>
using namespace std;

int main()
{
     string name;
     cout << "Please enter your full name" << endl;
     cin >> name;
     cout << “Hello, " << name << "! You are welcome" << endl;

     return 0;
}
```

3. Save the file with the name `read.cpp`.
4. Open the command prompt and change to the directory where you saved the file.
5. Compile the file using g++ by issuing the command `g++ read.cpp -o read.out`.
6. Run the code using the command `read.out`. and enter your full name as the input.

Observation:

Do you see your full name in the output?

Step 2: Reading an entire line

- **Objective**: Learn how to read the entire line.

Instructions:

1. Type in the following code as a new file:

```
#include <iostream>
#include <string>
using namespace std;

   int main()
   {
        string name;
        cout << "Please enter your full name" << endl;
        getline(cin, name);
        cout << “Hello, " << name << "! You are welcome" << endl;

        return 0;
   }
```

2. Save the file with the name `readline.cpp`.
3. Compile the file using g++ by issuing the command
   `g++ readline.cpp -o readline.out`.
4. Run the code using the command `readline.out` and enter your full name as the input.

Observation:

1. Do you see your full name in the output now?

Step 3: Opening a file for writing

- **Objective**: Learn how to open a file for writing.

Instructions:

1. Type in the following code as a new file:

```
#include <iostream>
#include <fstream>
using namespace std;

int main () {
     ofstream ofstr;
     ofstr.open ("contents.txt");
     ofstr << "First line of the file.\n";
     ofstr.close();

     return 0;
}
```

2. Save the file as `create_file.cpp`.
3. Compile and run the program.
4. Locate the file `contents.txt` using Windows Explorer, open it in a text editor, and see the contents of the file.
5. Change the message in the 8th line to "`Second line of the file`" and save the file.
6. Compile and run the program.
7. Locate the file `contents.txt` using Windows Explorer, open it in a text editor, and see the contents of the file.

Observation:

1. Do you see two lines in the file now?
2. What happened to the earlier text?

Step 4: Opening a file for writing

- **Objective**: Learn how to open a file for appending.

Instructions:

1. Type in the following code as a new file:

```
#include <iostream>
#include <fstream>
using namespace std;

int main () {
     ofstream ofstr;
     ofstr.open ("contents.txt", ios::app);
     ofstr << "Third line of the file.\n";
     ofstr.close();

     return 0;
}
```

2. Save the file as `append_file.cpp`.
3. Compile and run the program.
4. Locate the file `contents.txt` using Windows Explorer, open it in a text editor, and see the contents of the file.

Challenge:

Write a program to read lines of text from the command prompt continuously and write the input line-by-line into the text file `contents.txt` until a blank line is input.

Step 5: Opening the file for reading

- **Objective**: Learn how to open a file for reading its content.

Instructions:

1. Type in the following code as a new file:

```
#include <iostream>
#include <string>
#include <fstream>
using namespace std;

int main () {
     string line;
     ifstream ifstr;
     ifstr.open ("contents.txt");
     while (getline (ifstr, line)) {
          cout << line << endl;
     }
     ifstr.close();

     return 0;
}
```

2. Save the file as `read_file.cpp`.
3. Compile and run the program.

Post-Practical Reflection:

- **Reflection Questions**:

    - What are the different modes of accessing files you learnt during this practical?
    - What are the other modes of file access?
- **Peer Discussion**: discuss with your friends and list the key points that you learned.

Submission and Deadlines:

- **What to Submit**: submit the list of key points that you learnt as a hand-written document.
